

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-007847
 (43)Date of publication of application : 12.01.2001

(51)Int.CI.

H04L 12/56
 H04L 29/06

(21)Application number : 11-172462

(71)Applicant : NEC CORP

(22)Date of filing : 18.06.1999

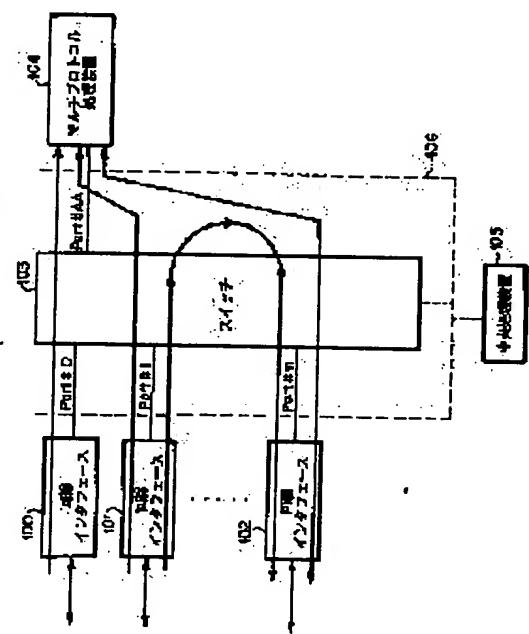
(72)Inventor : FUKUMOTO KEISUKE
 OKAMOTO TSUGIO

(54) MULTI-PROTOCOL PROCESSING DEVICE, CIRCUIT INTERFACE AND MULTI-PROTOCOL SWITCH SYSTEM HAVING THEM

(57)Abstract:

PROBLEM TO BE SOLVED: To improve efficiency in a port to which a protocol processing device is connected and to reduce the load on a protocol processor of the protocol processing device.

SOLUTION: Circuit interfaces 100, 101 or 102 references header information of received data and a forwarding table. In the case that the transmission destination of the received data is the circuit interface, the port number of a port to which the circuit interface is connected is added to the received data and the resulting data are outputted to a switch 103, and in the case that the transmission destination of the received data is a multi-protocol processing device 104, the port number of a port to which the multi-protocol processing unit 104 is connected and a processor identifier of a protocol processor handling the data are added to the received data and the resulting data are outputted to the switch 103. The multi-protocol processing device 104 uses the protocol processor to be identified by the processor identifier to process the data.



LEGAL STATUS

[Date of request for examination] 25.05.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3449294

[Date of registration] 11.07.2003

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-007847
 (43)Date of publication of application : 12.01.2001

(51)Int.Cl.

H04L 12/56
 H04L 29/06

(21)Application number : 11-172462

(71)Applicant : NEC CORP

(22)Date of filing : 18.06.1999

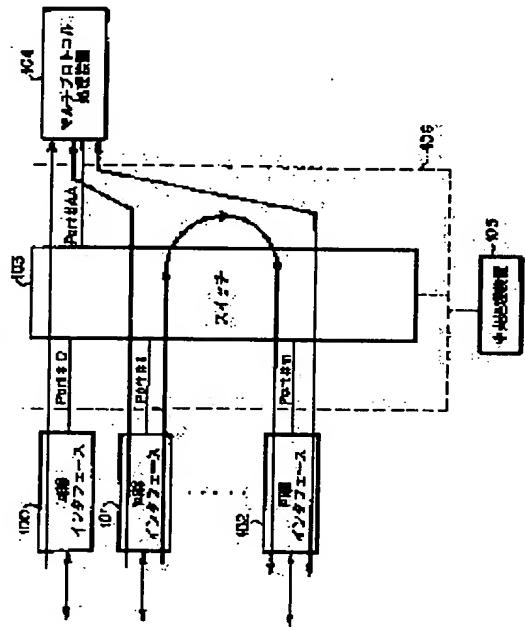
(72)Inventor : FUKUMOTO KEISUKE
 OKAMOTO TSUGIO

(54) MULTI-PROTOCOL PROCESSING DEVICE, CIRCUIT INTERFACE AND MULTI-PROTOCOL SWITCH SYSTEM HAVING THEM

(57)Abstract:

PROBLEM TO BE SOLVED: To improve efficiency in a port to which a protocol processing device is connected and to reduce the load on a protocol processor of the protocol processing device.

SOLUTION: Circuit interfaces 100, 101 or 102 references header information of received data and a forwarding table. In the case that the transmission destination of the received data is the circuit interface, the port number of a port to which the circuit interface is connected is added to the received data and the resulting data are outputted to a switch 103, and in the case that the transmission destination of the received data is a multi-protocol processing device 104, the port number of a port to which the multi-protocol processing unit 104 is connected and a processor identifier of a protocol processor handling the data are added to the received data and the resulting data are outputted to the switch 103. The multi-protocol processing device 104 uses the protocol processor to be identified by the processor identifier to process the data.



LEGAL STATUS

[Date of request for examination] 25.05.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3449294

[Date of registration] 11.07.2003

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-7847

(P2001-7847A)

(43)公開日 平成13年1月12日(2001.1.12)

(51)Int.Cl.⁷

H 04 L 12/56
29/06

識別記号

F I

テ-マコ-ト(参考)

H 04 L 11/20
13/00

1 0 2 A 5 K 0 3 0
3 0 5 C 5 K 0 3 4

審査請求 有 請求項の数9 OL (全8頁)

(21)出願番号 特願平11-172462

(22)出願日 平成11年6月18日(1999.6.18)

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 福元 啓介

東京都港区芝五丁目7番1号 日本電気株式会社内

(72)発明者 岡本 繼男

東京都港区芝五丁目7番1号 日本電気株式会社内

(74)代理人 100065385

弁理士 山下 積平

Fターム(参考) 5K030 HA08 HB18 JA05 JA08 KA13

LB02 LE02 LE03 LE06

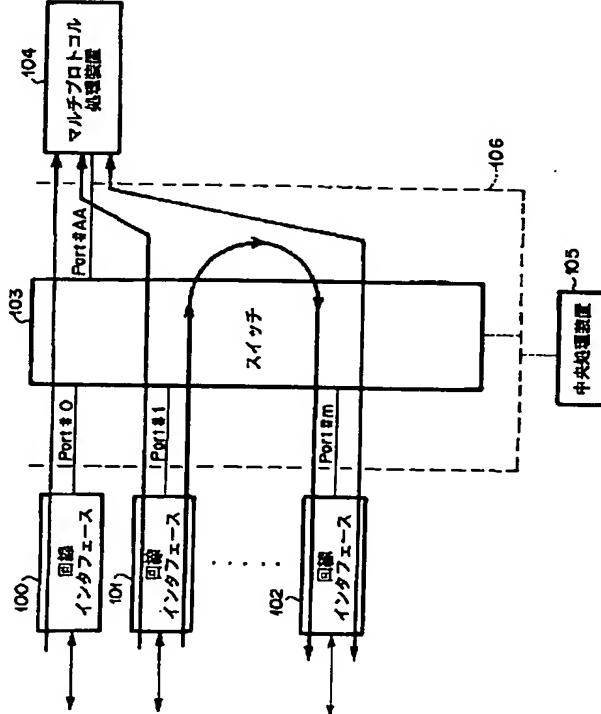
5K034 AA07 HH63 JJ24 MM06 QQ02

(54)【発明の名称】マルチプロトコル処理装置、回線インターフェース及びそれらを有するマルチプロトコルスイッチシステム

(57)【要約】

【課題】プロトコル処理装置が接続されているポートの使用効率を上げ、プロトコル処理装置のプロトコルプロセッサの処理負荷を低減する。

【解決手段】回線インターフェース100、101又は102は、受信データのヘッダ情報とフォワーディングテーブルを参照し、受信データの送信先が回線インターフェースである場合は、受信データにその回線インターフェースが接続されているポートのポート番号を付加してスイッチ103に出力し、受信データの送信先がマルチプロトコル処理装置104である場合は、受信データにマルチプロトコル処理装置104が接続されているポートのポート番号とデータを扱うプロトコルプロセッサのプロセッサ識別子を付加してスイッチ103に出力する。マルチプロトコル処理装置104は、プロセッサ識別子で識別されるプロトコルプロセッサを用いてデータを処理する。



【特許請求の範囲】

【請求項1】 ネットワークからデータを入力する入力手段と、フォワーディングテーブルと、前記入力したデータのヘッダの内容と前記フォワーディングテーブルの内容をもとに入力したデータの転送先を決定する決定部と、前記決定部の決定した転送先がプロトコルプロセッサであれば、転送先のプロトコルプロセッサのマルチプロトコル処理装置内のプロセッサ識別子と前記プロトコルプロセッサを備えるマルチプロトコル処理装置が接続されるポート番号を前記入力したデータに付加することにより、前記決定部の決定した転送先が回線インターフェースであれば、転送先の回線インターフェースが接続されるポート番号を前記入力したデータに付加することによりパケットを生成するパケット生成部と、前記パケットをポートを介してスイッチに出力する出力部とを備える複数の回線インターフェースと、

前記パケットをそのパケットが有するポート番号を有するポートに転送するスイッチと、
パケットをポートを介して前記スイッチとの間で入出力するスイッチインターフェースと、プロトコルの種類毎に設けられ各々がマルチプロトコル処理装置内で定まるプロセッサ識別子を有する複数のプロトコルプロセッサと、前記スイッチインターフェースを介して入力したパケットをそのパケットが有するプロセッサ識別子を有するプロトコルプロセッサに出力するプロセッサ選択部とを有する1又は2以上のマルチプロトコル処理装置と、
を備え、前記複数の回線インターフェースの1の回線インターフェースの扱うプロトコルが前記複数の回線のインターフェースの他の1の回線インターフェースの扱うプロトコルと異なることを特徴とするマルチプロトコルスイッチシステム。

【請求項2】 請求項1に記載のマルチプロトコルスイッチにおいて、プロトコルプロセッサ識別子は、同一のプロトコルに対してマルチプロトコル処理装置間で共通であることを特徴とするマルチプロトコルスイッチシステム。

【請求項3】 請求項1又は2に記載のマルチプロトコルスイッチシステムにおいて、前記フォワーディングテーブルは、前記複数のマルチプロトコル処理装置のいずれかから前記複数の回線インターフェースの各々に送られることを特徴とするマルチプロトコルスイッチシステム。

【請求項4】 請求項1乃至3のいずれか1項に記載のマルチプロトコルスイッチシステムにおいて、前記プロセッサ識別子は、前記複数のマルチプロトコル処理装置の各々から前記複数の回線インターフェースの各々に送られることを特徴とするマルチプロトコルスイッチシステム。

【請求項5】 請求項1乃至4のいずれか1項に記載のマルチプロトコルスイッチシステムにおいて、前記複数

のマルチプロトコル処理装置のうちの少なくとも2つのマルチプロトコル処理装置は同一のプロトコルを扱うプロトコルプロセッサを備えることを特徴とするマルチプロトコルスイッチシステム。

【請求項6】 請求項5に記載のマルチプロトコルスイッチシステムにおいて、同一のプロトコルを扱うプロトコルプロセッサを静的に分散させるための手段を更に備えることを特徴とするマルチプロトコルスイッチシステム。

10 【請求項7】 請求項5に記載のマルチプロトコルスイッチシステムにおいて、同一のプロトコルを扱うプロトコルプロセッサを動的に変化させるための手段を更に備えることを特徴とするマルチプロトコルスイッチシステム。

【請求項8】 ネットワークからデータを入力する入力手段と、フォワーディングテーブルと、前記入力したデータのヘッダの内容と前記フォワーディングテーブルの内容をもとに入力したデータの転送先を決定する決定部と、前記決定部の決定した転送先がプロトコルプロセッサであれば、転送先のプロトコルプロセッサのマルチプロトコル処理装置内のプロセッサ識別子と前記プロトコルプロセッサを備えるマルチプロトコル処理装置が接続されるポート番号を前記入力したデータに付加することにより、前記決定部の決定した転送先が回線インターフェースであれば、転送先の回線インターフェースが接続されるポート番号を前記入力したデータに付加することによりパケットを生成するパケット生成部と、前記パケットをポートを介してスイッチに出力する出力部とを備える回線インターフェース。

20 【請求項9】 パケットをポートを介してスイッチとの間で入出力するスイッチインターフェースと、プロトコルの種類毎に設けられ各々がマルチプロトコル処理装置内で定まるプロセッサ識別子を有する複数のプロトコルプロセッサと、前記スイッチインターフェースを介して入力したパケットをそのパケットが有するプロセッサ識別子を有するプロトコルプロセッサに出力するプロセッサ選択部とを有するマルチプロトコル処理装置。

【発明の詳細な説明】

【0001】

40 【発明の属する技術分野】 本発明は、ATM(Asynchronous Transfer Mode)、イーサネット(登録商標)等の多種多様なインターフェイスを収容して、その上位のプロトコルを処理するマルチプロトコルスイッチシステム、そのシステムに備わり回線とのインターフェースを行う回線インターフェース及びそのシステムそのトランク部に収容されるマルチプロトコル処理装置に関する。

【0002】

【従来の技術】 従来のマルチプロトコルスイッチシステムでは、回線インターフェースの種類に応じて、回線インターフェースの各々のプロトコルを処理するプロトコル処

理装置をスイッチのトランク部に複数種類搭載する必要があった。

【0003】

【発明が解決しようとする課題】通常、マルチプロトコルスイッチシステムにおいて、回線インタフェース間でのデータ転送のトラフィックが、プロトコル処理装置への、又は、からのトラフィックより多い。つまり、プロトコル処理装置-スイッチ間のバスの使用率は、回線インタフェース-スイッチ間の使用率より低いため、複数のプロトコル処理装置をマルチプロトコルスイッチシステムに搭載すると、複数のプロトコル処理装置に接続される使用率の低いポートが複数存在することになりスイッチの資源を効率良く使えないという問題があった。

【0004】また、複数のプロトコルを1つのプロセッサで処理すると、そのプロセッサに負荷が掛かりすぎてしまうという問題もあった。

【0005】更に、マルチプロトコルスイッチシステムではデータを回線インターフェースから回線インターフェースに中継する機能以外に、中継先の回線インターフェースを決定するルーティング先決定機能が必要である。また、マルチプロトコルスイッチシステムの輻輳状態や集中処理部の処理状態を中央処理装置が管理して、それらの状態を各回線インターフェースに通知することにより、中継を担う集中処理部をダイナミックに変更する機能が必要である。これらの3つの機能はデータを入力する回線インターフェースのみの情報だけでは実現できない。これらの機能を実現するためには、全ての回線インターフェースからの情報を用いた集中的な処理が必要である。

【0006】更に、回線インターフェイスはATMのようなコネクションオリエンティッド通信の回線インターフェイス及びIPを代表とするコネクションレス通信の回線インターフェイス等、多種多様にわたる。上記のルーティング処理を例にとって考えると、コネクションオリエンティッド通信のルーティング処理及びコネクションレス通信のルーティング処理の両方を集中的に行なうことは処理負荷が大きい。また、回線インターフェイスの種類を増やそうとした場合等に、前述の集中的な処理を行なう部分の負荷が大きく、回線インターフェイスの種類や数の追加が困難な場合も考えられる。

【0007】更に、回線インターフェイスから集中制御部へデータを転送する際に、回線インターフェイスと集中制御部を従来のようなバスで接続すると、回線インターフェイスの増大に対して、転送能力が足らなくなる。

【0008】本発明は上記の従来の技術の問題点に鑑みてなされたもので、プロトコル処理装置が接続されるポートの使用効率が高いマルチプロトコルスイッチシステムを提供することを目的とする。

【0009】また本発明は、プロトコルプロセッサの処理負荷が低減されたマルチプロトコルスイッチシステム

を提供することを目的とする。

【0010】

【課題を解決するための手段】本発明によるマルチプロトコルスイッチシステムは、ネットワークからデータを入力する入力手段と、フォワーディングテーブルと、前記入力したデータのヘッダの内容と前記フォワーディングテーブルの内容をもとに入力したデータの転送先を決定する決定部と、前記決定部の決定した転送先がプロトコルプロセッサであれば、転送先のプロトコルプロセッサのマルチプロトコル処理装置内のプロセッサ識別子と前記プロトコルプロセッサを備えるマルチプロトコル処理装置が接続されるポート番号を前記入力したデータに付加することにより、前記決定部の決定した転送先が回線インターフェースであれば、転送先の回線インターフェースが接続されるポート番号を前記入力したデータに付加することによりパケットを生成するパケット生成部と、前記パケットをポートを介してスイッチに出力する出力部とを備える複数の回線インターフェースと、前記パケットをそのパケットが有するポート番号を有するポートに転送するスイッチと、パケットをポートを介して前記スイッチとの間で入出力するスイッチインターフェースと、プロトコルの種類毎に設けられ各々がマルチプロトコル処理装置内で定まるプロセッサ識別子を有する複数のプロトコルプロセッサと、前記スイッチインターフェースを介して入力したパケットをそのパケットが有するプロセッサ識別子を有するプロトコルプロセッサに出力するプロセッサ選択部とを有する1又は2以上のマルチプロトコル処理装置と、を備え、前記複数の回線インターフェースの1の回線インターフェースの扱うプロトコルが前記複数の回線のインターフェースの他の1の回線インターフェースの扱うプロトコルと異なることを特徴とする。

【0011】また、本発明によるマルチプロトコルスイッチシステムは、上記のマルチプロトコルスイッチにおいて、プロトコルプロセッサ識別子は、同一のプロトコルに対してマルチプロトコル処理装置間で共通であることを特徴とする。

【0012】更に、本発明によるマルチプロトコルスイッチシステムは、上記のマルチプロトコルスイッチシステムにおいて、前記フォワーディングテーブルは、前記複数のマルチプロトコル処理装置のいずれかから前記複数の回線インターフェースの各々に送られることを特徴とする。

【0013】更に、本発明によるマルチプロトコルスイッチシステムは、上記のマルチプロトコルスイッチシステムにおいて、前記プロセッサ識別子は、前記複数のマルチプロトコル処理装置の各々から前記複数の回線インターフェースの各々に送られることを特徴とする。

【0014】更に、本発明によるマルチプロトコルスイッチシステムは、上記のマルチプロトコルスイッチシステムにおいて、前記複数のマルチプロトコル処理装置の

うちの少なくとも2つのマルチプロトコル処理装置は同一のプロトコルを扱うプロトコルプロセッサを備えることを特徴とする。

【0015】更に、本発明によるマルチプロトコルシステムは、上記のマルチプロトコルシステムにおいて、同一のプロトコルを扱うプロトコルプロセッサを静的に分散させるための手段を更に備えることを特徴とする。

【0016】更に、本発明によるマルチプロトコルシステムは、上記のマルチプロトコルシステムにおいて、同一のプロトコルを扱うプロトコルプロセッサを動的に変化させるための手段を更に備えることを特徴とする。

【0017】本発明による回線インターフェースは、ネットワークからデータを入力する入力手段と、フォワーディングテーブルと、前記入力したデータのヘッダの内容と前記フォワーディングテーブルの内容をもとに入力したデータの転送先を決定する決定部と、前記決定部の決定した転送先がプロトコルプロセッサであれば、転送先のプロトコルプロセッサのマルチプロトコル処理装置内でのプロセッサ識別子と前記プロトコルプロセッサを備えるマルチプロトコル処理装置が接続されるポート番号を前記入力したデータに付加することにより、前記決定部の決定した転送先が回線インターフェースであれば、転送先の回線インターフェースが接続されるポート番号を前記入力したデータに付加することによりパケットを生成するパケット生成部と、前記パケットをポートを介してスイッチに出力する出力部とを備える。

【0018】本発明によるマルチプロトコル処理装置は、パケットをポートを介してスイッチとの間で入出力するスイッチインターフェースと、プロトコルの種類毎に設けられ各々がマルチプロトコル処理装置内で定まるプロセッサ識別子を有する複数のプロトコルプロセッサと、前記スイッチインターフェースを介して入力したパケットをそのパケットが有するプロセッサ識別子を有するプロトコルプロセッサに出力するプロセッサ選択部とを有する。

【0019】

【発明の実施の形態】本発明によるマルチプロトコルシステムは、ATM(Asynchronous Transfer Mode)、イーサネット等の多種多様なインターフェイスを収容して、その上位のプロトコルを処理するマルチプロトコルシステムにおいて、スイッチの性能を最大限に使用するためにスイッチのトランク部に複数種類のプロトコルを処理するマルチプロトコル処理装置を搭載し、かつ、マルチプロトコル処理装置はプロトコル毎に分散して処理する機能を持つことを特長とする。また、本発明によるマルチプロトコルシステムは、プロトコル毎の処理の振り分けをセルフルーティングで行うこととする。

【0020】図2を参照すると、回線インターフェース100、101又は102の入力部A201は、マルチプロトコルスイッチシステムで中継されるべきデータをネットワークから受信する。決定部203は、回線インターフェースの内部のフォワーディングテーブル202より検索された内容及びデータのヘッダ情報をもとにデータの転送先が接続されているポート番号を決定する。パケット生成部204は、データの転送先がマルチプロトコル処理装置104である場合には、データに、マルチプロトコル処理装置104が接続されているポートのポート番号と、データを処理するべきプロトコルプロセッサのマルチプロトコル処理装置104内の識別子(以下、「プロセッサID」という)を付加する。また、パケット生成部204は、データの転送先が回線インターフェースである場合には、データにその回線インターフェースが接続されているポートのポート番号を付加する。出力部205は、パケット生成部204が生成したパケットをスイッチ103に出力する。

【0021】図1を参照すると、スイッチ103は、回線インターフェース100、101又は102より入力したデータを、決定されたポート番号で指定されるポートに転送する。

【0022】図2を参照すると、回線インターフェース100、101又は102の入力部B206は、スイッチ103からポートを経由してパケットを受信する。ポート番号削除部207はパケットからポート番号を削除する。ヘッダ付加部208は、フォワーディングテーブル202から、次に転送する端末のデータリンク層のアドレスを得て、データリンク層のヘッダであってそのアドレスを有するものを再構築する。出力部Bは再構築されたヘッダを有するデータをネットワークに出力する。

【0023】本発明では、スイッチのトランク部に集中制御部としてのマルチプロトコル処理装置104を収容し、バスではなく、スイッチを用いて、回線インターフェイスとマルチプロトコル処理装置を接続する。これにより、従来のバスの転送能力の問題を解決する。

【0024】また、マルチプロトコル処理装置104において、ATM、IP(InternetProtocol)等のプロトコルを処理するプロトコルプロセッサをそのプロトコル毎に配備することでプロトコル単位の分散処理を行い、処理の集中の問題を解決する。

【0025】更に、スイッチ103と複数のプロトコルプロセッサのいずれかのプロトコルプロセッサをセルフルーティングにより接続することで、プロトコル処理単位のデータの振り分けを高速化して、処理能力の向上を図る。以下にその例を示す。

【0026】回線インターフェース100は、ルーティングプロトコルの処理等を必要なデータを受信したことを見出されたデータのヘッダ情報をより知った場合、スイッチ103を介して、その受信データをマルチプロトコル処理

装置104に転送する。その際、回線インターフェースは、マルチプロトコル処理装置104が接続されているスイッチのポート番号と受信データに必要なプロトコル処理をするプロトコルプロセッサのプロセッサIDを受信データに付加して指定してスイッチ103に出力する。

【0027】図3を参照すると、マルチプロトコル処理装置104のスイッチインターフェース303は、スイッチ103からポートを介してパケットを受信すると、パケットからポート番号を削除してプロセッサ選択部304にデータを渡す。プロセッサ選択部304は、プロセッサIDからデータを引き渡すプロトコルプロセッサを判断し、パケットからプロセッサIDを削除してデータをプロトコルプロセッサに渡す。プロトコルプロセッサ305～307のいずれかはポート番号とプロセッサIDによりセルフルーティングされたデータを受信すると、事前に自らに与えられたプロトコル処理をする。プロトコル処理の例としては、ATMセルの場合であればシグナリング、IPパケットの場合であればルーティング計算などのプロトコル処理がある。

【0028】図1は、本発明の実施形態によるマルチプロトコルスイッチシステムの構成を示すブロック図である。図1において100～102は、回線インターフェース、103はスイッチ、104はマルチプロトコル処理装置、105は中央処理装置、106は中央処理装置105が回線インターフェース100～102やマルチプロトコル処理装置104の初期設定や状態監視を行うための保守バスである。スイッチ103と回線インターフェース100～102はポートを介して接続され、スイッチ103とマルチプロトコル処理装置104もポートを介して接続される。

【0029】マルチプロトコル処理装置104は、スイッチのトランク部に収容される。中央処理装置105は装置管理等の制御を行う。通常の場合、中央処理部105は、プロセッサ、メインメモリ、2次記憶装置から構成され、メインメモリに記憶されるプログラムに従って動作する。

【0030】回線インターフェース100～102は、ATM、イーサネット、T1/E1などのインターフェースを持つ回線インターフェースカードである。例えはイーサネットのインターフェースを持つ回線インターフェースカードであれば、IPアドレスから転送先のスイッチのポート番号を決定し、内部ヘッダとしてスイッチのポート番号をイーサフレームに付加してスイッチ103に出力する。また、スイッチ103からデータを受信した回線インターフェース100～102は、ポート番号を削除し、IPアドレスをキーにフォワーディングテーブルを検索して次に転送する端末のMACアドレスを得て、イーサフレームのヘッダを再構築してから、データをネットワークに出力する。スイッチ103は、スイッチポ

ート番号にしたがって、データをスイッチングにより転送する。マルチプロトコル処理装置104は、ルーティング機能等の処理を行い、種類の異なる各回線インターフェース100～102からスイッチ103を介して入力されたデータに対して各プロトコルに応じた処理を行う。例えば、ATMセルであればシグナリング処理を行い、IPパケットであればルーティングプロトコルのパケットを受信して、ルーティング計算を行い、最新のルーティングテーブルを作成して、IPレイヤでの転送を要求されるインターフェースを持つ回線インターフェースにフォワーディングテーブルを配布するなどの処理を行う。

【0031】図1においてマルチプロトコル処理装置の数は一台であるが、複数台搭載してもよい。

【0032】図3はマルチプロトコル処理装置104のブロック図である。

【0033】図3を参照すると、マルチプロトコル処理装置104は、スイッチ103とのインターフェース部303、プロセッサ選択部304、各回線インターフェース100～102からスイッチ103を介して入力されたデータを処理するプロトコルプロセッサ部305～307、メインメモリ302、マルチプロトコル処理装置104全体の管理を行うCPU301から構成される。スイッチインターフェース303は、スイッチ103から受信したパケットを受信したデータの順番にプロセッサ選択部304に出力する。プロセッサ選択部304は、スイッチインターフェース303から受け取ったパケット内のデータをどのプロトコルプロセッサに渡せば良いかをプロセッサIDにより判断する。プロトコルプロセッサ305～307の種類の数は、回線インターフェースの種類の数であり、プロトコルプロセッサ305～307のそれぞれは、ATMのプロトコルプロセッサ、IPパケットのプロトコルプロセッサなどである。

【0034】図1を参照して本実施形態の動作について詳細に説明する。

【0035】図1において回線インターフェース100、101又は102がATMのインターフェースを持つ回線インターフェースであった場合を例に説明する。ATMセルを受信した回線インターフェース100、101又は102は、ATMヘッダのVC1(Virtual Channel Identifier)から転送先のポート番号を決定し、内部ヘッダとしてポート番号をATMセルに付加してスイッチ103に出力する。なお、転送先は回線インターフェース100、101又は102がスイッチ103からデータを受信するとポート番号を削除し、ATMヘッダのVC1からそれに対応するVPI(Virtual Path Identifier)を割り当ててネットワークにATMセルを出力する。

【0036】上記処理において、ATM網がPVC(Per

manent Virtual Connection)の場合、VCIと転送先ポート番号の対応付け、VCIとVPIの対応付けは、マルチプロトコル処理装置104から配布されるフォワーディングテーブルを検索して行う。図4にフォワーディングテーブルの一例を示す。またスイッチ103に出力する時のフォーマットを図5に示す。図5を参照すると、フォーマットはATMセルに転送先ポート番号を附加したフォーマットである。

【0037】ATM網がSVC(Switched Virtual Connection)の場合、マルチプロトコル処理装置104にUNI(User-Network Interface)シグナリングやPNNI(Private Network Node Interface)シグナリングを実装してシグナリング処理を行い、呼の設定や解放などの処理を行う。

【0038】図1において、転送先のポート番号がマルチプロトコル処理装置を搭載しているポート番号であった場合の動作について説明する。回線インタフェース100、101又は102は、シグナリング処理を要求するデータを構成するATMセルの場合、ATMセルにマルチプロトコル処理装置104を示す転送先ポート番号とプロセッサIDを附加してスイッチに出力する。この時のフォーマットを図6に示す。ポート番号は、スイッチ103がスイッチングするための情報であり、この場合は、マルチプロトコル処理装置104が搭載されているポート番号とする。図1においてはポート番号#AAを指定することになる。プロセッサIDは、マルチプロトコル処理装置104に搭載され、各プロトコルに応じた処理を行うプロトコルプロセッサを示す倫理的なIDである。マルチプロトコル処理装置104は、プロセッサIDを参照して、プロトコルプロセッサ305、306又は307にセルフルーティングする。

【0039】回線インタフェース100、101又は102が各マルチプロトコル処理装置の各プロトコルプロセッサのプロセッサIDであって、各プロトコルに応じたプロセッサIDを取得する方法は以下の通りである。システムの初期値設定時や回線インタフェースが追加された時に、マルチプロトコル処理装置104は、中央処理装置105から保守バス106経由で、各ポートに接続されている回線インタフェースがどのプロトコルを扱うのかについての情報をもらい、各回線インタフェースが扱うプロトコルに応じて、各回線インタフェースが使用するプロトコルを扱うプロトコルプロセッサのプロセッサIDを中央処理装置105に通知する。中央処理装置105が各回線インタフェースにプロセッサIDにマルチプロトコル処理装置のポート番号を附加して通知することにより、各回線インタフェースは自らが扱うプロトコルを処理するプロトコルプロセッサを有するマルチプロトコル処理装置のポート番号とそのプロトコルを扱うマルチプロトコルプロセッサのプロセッサIDを取得する。同じプロトコルを扱う複数の回線インタフェース

がスイッチ103に接続されていて、且つ、同じプロトコルを扱うプロトコルプロセッサを有する複数のマルチプロトコル処理装置がスイッチ103に接続されている場合は、異なった回線インタフェースに異なったポート番号とプロセッサIDを通知することにより、マルチプロトコル処理装置に搭載されるプロトコルプロセッサの処理の負荷を分散できる。これが静的な処理負荷分散であるが、以下のような動的な処理負荷分散もある。すなわち、マルチプロトコル処理装置104は、あるプロトコルプロセッサの処理の負荷が重くなったならば、そのプロトコルプロセッサのプロセッサIDを中央処理装置に送る。中央処理装置は、そのプロトコルプロセッサと同一のプロトコルを扱う他のマルチプロトコル処理装置内のプロトコルプロセッサのプロセッサIDとそのマルチプロトコル処理装置のポート番号を、そのプロトコルを扱う回線インタフェースに通知する。通知を受けた回線インタフェースは、データのルーティング先を切り替える。

【0040】ATMの回線インタフェースが上記の処理20で取得したプロセッサIDが#BBであれば、回線インタフェースがポート番号#AA、プロセッサID#BBをATMセルに附加することにより、ATMセルは、マルチプロトコル処理装置104に搭載されるATMプロトコルプロセッサまでセルフルーティングされ、ATMプロトコルプロセッサは、シグナリング処理などのプロトコル処理を行う。

【0041】図3を参照してマルチプロトコル処理装置104の動作について詳細に説明する。

【0042】スイッチインタフェース303は、図6に30示すフォーマットのパケットを受信すると、パケットからポート番号を削除し、スイッチ103から受信したパケットを受信順にプロセッサ選択部304に出力する。プロセッサ選択部304は、プロセッサIDからプロトコルプロセッサ305～307のうちのどのプロトコルプロセッサにデータを引き渡せばよいかを判断し、パケットからプロセッサIDを削除して、プロセッサIDで指示されたプロトコルプロセッサ305～307にデータを出力する。各プロトコルプロセッサは、受信したデータを解析してプロトコル処理を行う。

【0043】例えばATMのプロトコルプロセッサで行われる処理としては、シグナリング処理があげられる。また、障害などによりATM網の構成が変わった時に、最新のフォワーディングテーブルを作成して、各ATMの回線インタフェースに配布する処理もあげられる。

【0044】また、イーサネットのプロトコルプロセッサで行われる処理としては、ルーティングプロトコルの情報を受けたときにルーティング計算を行い、ルーティングテーブルを作成し、イーサネットの回線インタフェースにフォワーディングテーブルを配布する処理などがある。

【0045】このようして、マルチプロトコル処理装置104は、多種多様な回線インターフェースからのデータを複数のプロトコルプロセッサ305～307にセルフルーティングで振り分け、複数のプロトコルを処理することを実現する。

【0046】プロトコルプロセッサで処理されたデータを回線インターフェースを通じて外部に出力する場合には、プロトコルプロセッサで処理されたデータには、回線インターフェースが接続されるポートのポート番号が付加されてパケット化され、生成されたパケットはスイッチ103に出力される。また、プロトコルプロセッサで処理されたデータが他のマルチプロトコル処理装置のプロトコルプロセッサで更に処理される必要がある場合には、プロトコルプロセッサで処理されたデータには、そのマルチプロトコル処理装置の接続されるポートのポート番号と、そのプロトコルプロセッサのプロセッサIDが付加されてパケット化され、生成されたパケットはスイッチ103に出力される。

【0047】上記の説明では、同一のプロトコルを扱うプロセッサであっても異なったマルチプロトコル処理装置に組み込まれていれば、そのプロセッサIDは異なるとしたが、同一のプロトコルを扱うプロセッサであれば異なったマルチプロトコル処理装置に組み込まれていても、そのプロセッサIDは同一であるとしても良い。

【0048】また、1つのマルチプロトコル処理装置に同一のプロトコルを扱う複数のプロトコルプロセッサが組み込まれていても良い。

【0049】

【発明の効果】以上説明したように、本発明によれば、スイッチのトランク部に複数のプロトコルを処理するマルチプロトコル処理装置を搭載することにより、プロトコル処理装置を複数搭載する必要がなく、また、単一のプロトコル処理装置に複数のプロトコルプロセッサを搭載するので、プロトコル処理装置とスイッチとの間のポートの使用率を向上することができる。更に、各プロトコル毎にプロトコルプロセッサを持つので、負荷を分散*

*することができる。

【図面の簡単な説明】

【図1】本発明の実施形態によるマルチプロトコルシステムの構成を示すブロック図である。

【図2】図1に示す回線インターフェースの構成を示すブロック図である。

【図3】図1に示すマルチプロトコルプロセッサの構成を示すブロック図である。

【図4】本発明の実施形態によるフォワーディングテーブルの例である。

【図5】従来例による回線インターフェースがOutputするATMセルをもととするパケットのフォーマット図である。

【図6】本発明の実施形態による回線インターフェースがOutputするATMセルをもととするパケット5のフォーマット図である。

【符号の説明】

100、101、102 回線インターフェース

103 スイッチ

104 マルチプロトコル処理装置

105 中央処理装置

201 入力部A

202 フォワーディングテーブル

203 決定部

204 パケット生成部

205 出力部A

206 入力部B

207 ポート番号削除部

208 ヘッダ付加部

209 出力部B

301 CPU

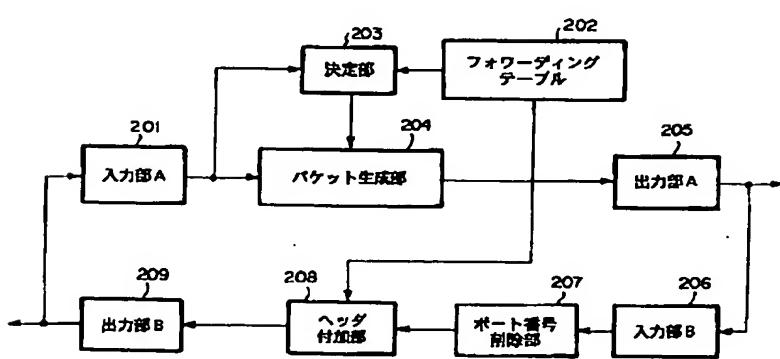
302 メインメモリ

303 スイッチインターフェース

304 プロセッサ選択部

305、306、307 プロトコルプロセッサ

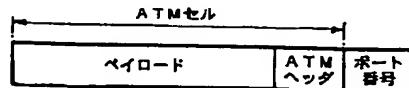
【図2】



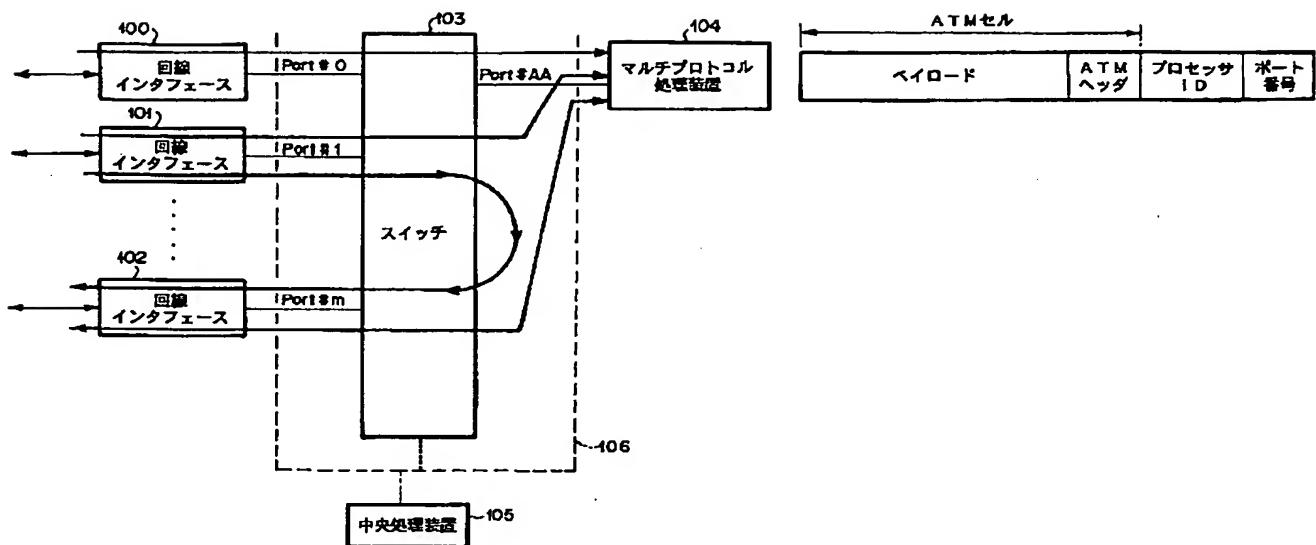
【図4】

VCI	ポート番号	VPI
11	1	4
20	5	7
13	3	14
⋮	⋮	⋮

【図5】



【図1】



【図6】

【図3】

